

16.11.2020



Laboratorium wprowadzające

Laboratorium 2

Karol Hamielec

Zadanie 1

1. Znajdź definicję struktury `file_operations`. Zapoznaj się z jej polami.

```
1814 struct file_operations {
1815     struct module *owner;
1816     loff_t (*llseek) (struct file *, loff_t, int);
1817     ssize_t (*read) (struct file *, char __user *, size_t, loff_t *);
1818     ssize_t (*write) (struct file *, const char __user *, size_t, loff_t *);
1819     ssize_t (*read_iter) (struct kiocb *, struct iov_iter *);
1820     ssize_t (*write_iter) (struct kiocb *, struct iov_iter *);
1821     int (*iopoll)(struct kiocb *kiocb, bool spin);
1822     int (*iterate) (struct file *, struct dir_context *);
1823     int (*iterate_shared) (struct file *, struct dir_context *);
1824     __poll_t (*poll) (struct file *, struct poll_table_struct *);
1825     long (*unlocked_ioctl) (struct file *, unsigned int, unsigned long);
1826     long (*compat_ioctl) (struct file *, unsigned int, unsigned long);
1827     int (*mmap) (struct file *, struct vm_area_struct *);
1828     unsigned long mmap_supported_flags;
1829     int (*open) (struct inode *, struct file *);
1830     int (*flush) (struct file *, fl_owner_t id);
1831     int (*release) (struct inode *, struct file *);
1832     int (*fsync) (struct file *, loff_t, loff_t, int datasync);
1833     int (*fasync) (int, struct file *, int);
1834     int (*lock) (struct file *, int, struct file_lock *);
1835     ssize_t (*sendpage) (struct file *, struct page *, int, size_t, loff_t *, int);
1836     unsigned long (*get_unmapped_area)(struct file *, unsigned long, unsigned long, unsigned long, unsigned long);
1837     int (*check_flags)(int);
1838     int (*flock) (struct file *, int, struct file_lock *);
1839     ssize_t (*splice_write)(struct pipe_inode_info *, struct file *, loff_t *, size_t, unsigned int);
1840     ssize_t (*splice_read)(struct file *, loff_t *, struct pipe_inode_info *, size_t, unsigned int);
1841     int (*setlease)(struct file *, long, struct file_lock **, void **);
1842     long (*fallocate)(struct file *file, int mode, loff_t offset,
1843                     loff_t len);
1844     void (*show_fdinfo)(struct seq_file *m, struct file *f);
1845 #ifdef CONFIG_MMU
1846     unsigned (*mmap_capabilities)(struct file *);
1847 #endif
1848     ssize_t (*copy_file_range)(struct file *, loff_t, struct file *,
1849                             loff_t, size_t, unsigned int);
1850     loff_t (*remap_file_range)(struct file *file_in, loff_t pos_in,
1851                             struct file *file_out, loff_t pos_out,
1852                             loff_t len, unsigned int remap_flags);
1853     int (*fadvise)(struct file *, loff_t, loff_t, int);
1854 } __randomize_layout;
```

strukturę udało mi się znaleźć w pliku `/linux/include/linux/fs.h`

2. Spróbuj odnaleźć użycie tej struktury w systemie plików `hostfs`.

```
382 static const struct file_operations hostfs_file_fops = {
383     .llseek      = generic_file_llseek,
384     .splice_read = generic_file_splice_read,
385     .read_iter   = generic_file_read_iter,
386     .write_iter  = generic_file_write_iter,
387     .mmap        = generic_file_mmap,
388     .open        = hostfs_open,
389     .release     = hostfs_file_release,
390     .fsync       = hostfs_fsync,
391 };
392
393 static const struct file_operations hostfs_dir_fops = {
394     .llseek      = generic_file_llseek,
395     .iterate_shared = hostfs_readdir,
396     .read        = generic_read_dir,
397     .open        = hostfs_open,
398     .fsync       = hostfs_fsync,
399 };
400
```

3. Czy potrafisz zinterpretować zastosowanie poszczególnych pól?

Dokonyjemy tutaj zmapowania poszczególnych wskaźników na funkcje na konkretne funkcje. Inne funkcje będą obsługiwać normalny plik, a inne będą obsługiwać foldery.

- lseek - ustawienia kursora w pliku
- splice - czytanie z pliku określonej ilości bajtów i przekazywanie tego do potoku
- read_iter - generyczna procedura do czytania z pliku
- write_iter - generyczna procedura do czytania z pliku
- mmap - mapowanie zawartości pliku do przestrzeni adresowej
- release - zapisanie mapowania do pliku
- fsync - zrzuci zawartość cache/bufora do urządzenia

4. Zlokalizuj w dokumentacji (katalog Documentation lub wyszukiwarka) informacje o tym, jak przechodzić po strukturze katalogów w jądrze.

5. Zlokalizuj plik namei.h a w nim funkcje kern_path i user_path. Czym się one różnią i kiedy mają zastosowanie?

```
2430 int kern_path(const char *name, unsigned int flags, struct path *path)
2431 {
2432     return filename_lookup(AT_FDCWD, getname_kernel(name),
2433                           flags, path, NULL);
2434 }
2435 EXPORT_SYMBOL(kern_path);
2436
2602
2603 int user_path_at_empty(int dfd, const char __user *name, unsigned flags,
2604                       struct path *path, int *empty)
2605 {
2606     return filename_lookup(dfd, getname_flags(name, flags, empty),
2607                           flags, path, NULL);
2608 }
```

If

Funkcja kern_path oraz user_path dokonuje wyszukiwania pliku, aby uzyskać inode. W funkcji kern_path wyszukiwanie dokonuje się relatywnie do aktualnego folderu roboczego procesu. Dodatkowo funkcje różnią się wywołaniem getname(), które służy do kopiowania nazwy pliku z przestrzeni adresowej procesu do przestrzeni adresowej jądra

6. Znajdź definicję struktury dentry.

```
89 struct dentry {
90     /* RCU lookup touched fields */
91     unsigned int d_flags;          /* protected by d_lock */
92     seqcount_t d_seq;             /* per dentry seqlock */
93     struct hlist_bl_node d_hash;  /* lookup hash list */
94     struct dentry *d_parent;      /* parent directory */
95     struct qstr d_name;
96     struct inode *d_inode;        /* Where the name belongs to - NULL is
97                                   * negative */
98     unsigned char d_iname[DNAME_INLINE_LEN]; /* small names */
99
100     /* Ref lookup also touches following */
101     struct lockref d_lockref;     /* per-dentry lock and refcount */
102     const struct dentry_operations *d_op;
103     struct super_block *d_sb;     /* The root of the dentry tree */
104     unsigned long d_time;         /* used by d_revalidate */
105     void *d_fsdata;              /* fs-specific data */
106
107     union {
108         struct list_head d_lru;    /* LRU list */
109         wait_queue_head_t *d_wait; /* in-lookup ones only */
110     };
111     struct list_head d_child;     /* child of parent list */
112     struct list_head d_subdirs;   /* our children */
113     /*
114      * d_alias and d_rcu can share memory
115      */
116     union {
117         struct hlist_node d_alias; /* inode alias list */
118         struct hlist_bl_node d_in_lookup_hash; /* only for in-lookup ones */
119         struct rcu_head d_rcu;
120     } d_u;
121 } __randomize_layout;
```

7. Znajdź funkcję dget. Co ona robi i po co? (Podpowiedź można znaleźć we wcześniej znalezionej dokumentacji.)

Zwraca wskaźnik na strukturę dentry oraz inkrementuje licznik referencji

Zadanie 2

1. Zapoznaj się z plikiem .config.

2. Co robi polecenie make oldconfig? Wykonaj je.

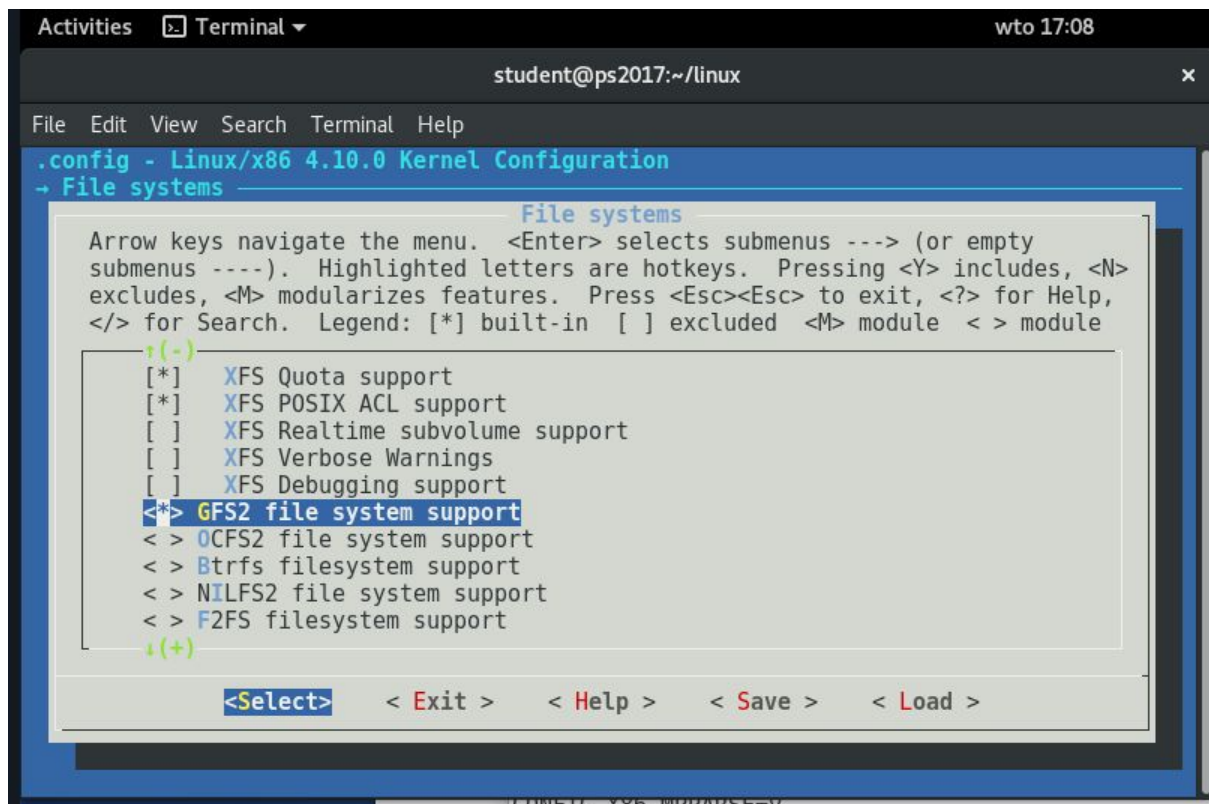
Aktualizuje aktualny config używając pliku .config jako bazy

3. Co robi polecenie make menuconfig? Wykonaj je.

Aktualizuje aktualny config za pomocą interfejsu menu

4. Ustaw dowolną, ale charakterystyczną wersję lokalną (CONFIG_LOCALVERSION albo General setup/Local version).

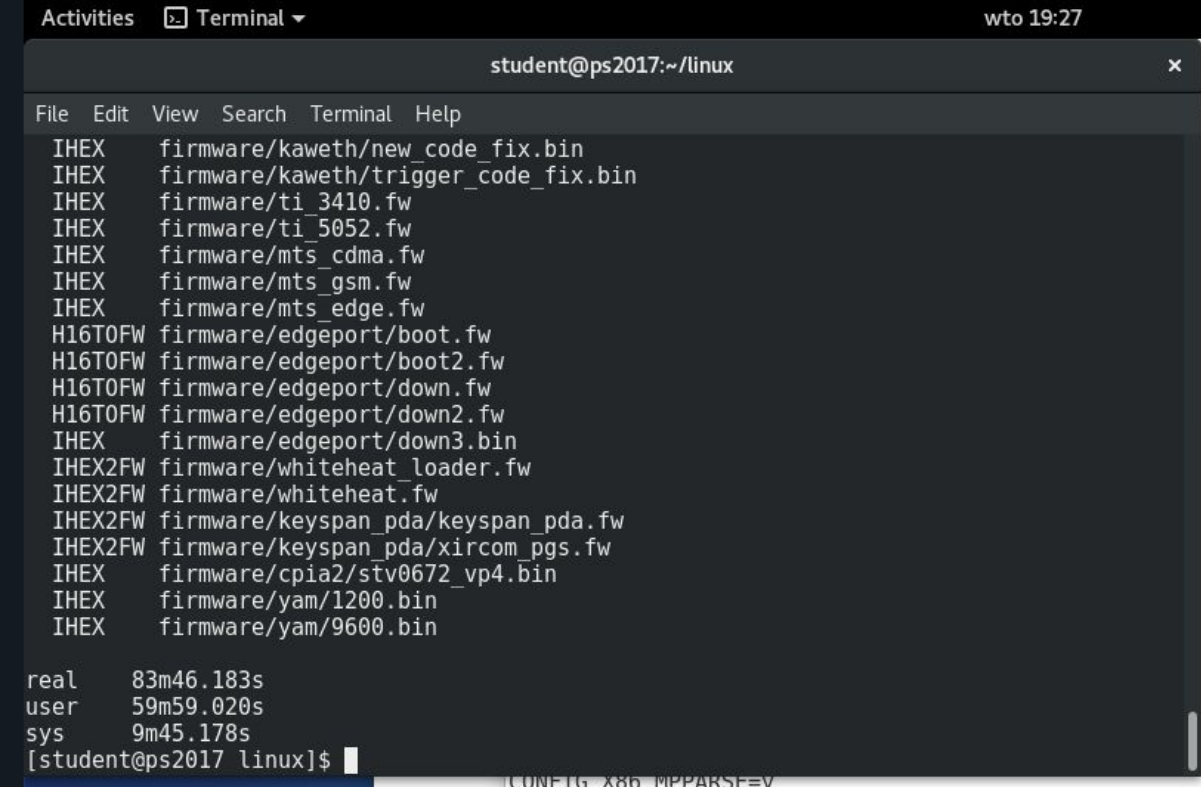
5. Zrób małą zmianę w konfiguracji (np. włącz obsługę któregoś systemu plików).



3. Co robi polecenie make all?

Buduje kernel oraz moduły

4. Skompiluj jądro po zmianie i zmierz czas kompilacji (komenda `time`).
Jakie informacje wyświetlają się podczas kompilacji?



```
Activities  Terminal  wto 19:27
student@ps2017:~/linux
File Edit View Search Terminal Help
IHEX      firmware/kaweth/new_code_fix.bin
IHEX      firmware/kaweth/trigger_code_fix.bin
IHEX      firmware/ti_3410.fw
IHEX      firmware/ti_5052.fw
IHEX      firmware/mts_cdma.fw
IHEX      firmware/mts_gsm.fw
IHEX      firmware/mts_edge.fw
H16T0FW   firmware/edgeport/boot.fw
H16T0FW   firmware/edgeport/boot2.fw
H16T0FW   firmware/edgeport/down.fw
H16T0FW   firmware/edgeport/down2.fw
IHEX      firmware/edgeport/down3.bin
IHEX2FW   firmware/whiteheat_loader.fw
IHEX2FW   firmware/whiteheat.fw
IHEX2FW   firmware/keyspan_pda/keyspan_pda.fw
IHEX2FW   firmware/keyspan_pda/xircom_pgs.fw
IHEX      firmware/cpia2/stv0672_vp4.bin
IHEX      firmware/yam/1200.bin
IHEX      firmware/yam/9600.bin

real      83m46.183s
user      59m59.020s
sys       9m45.178s
[student@ps2017 linux]$
```

5. Co robią polecenia `make modules_install` i `make install`?

`make modules` - zbudowanie modułów

`make install` - zainstalowanie modułów zbudowanych przy pomocy *make modules*

6. Zainstaluj jądro w systemie.

```
Activities Terminal wto 19:33
student@ps2017:~/linux
File Edit View Search Terminal Help
fdimage288 - Create 2.8MB boot floppy image (arch/x86/boot/fdimage)
isoimage - Create a boot CD-ROM image (arch/x86/boot/image.iso)
           bzdisk/fdimage*/isoimage also accept:
           FDARGS="..." arguments for the booted kernel
           FDINITRD=file initrd for the booted kernel

i386_defconfig - Build for i386
x86_64_defconfig - Build for x86_64

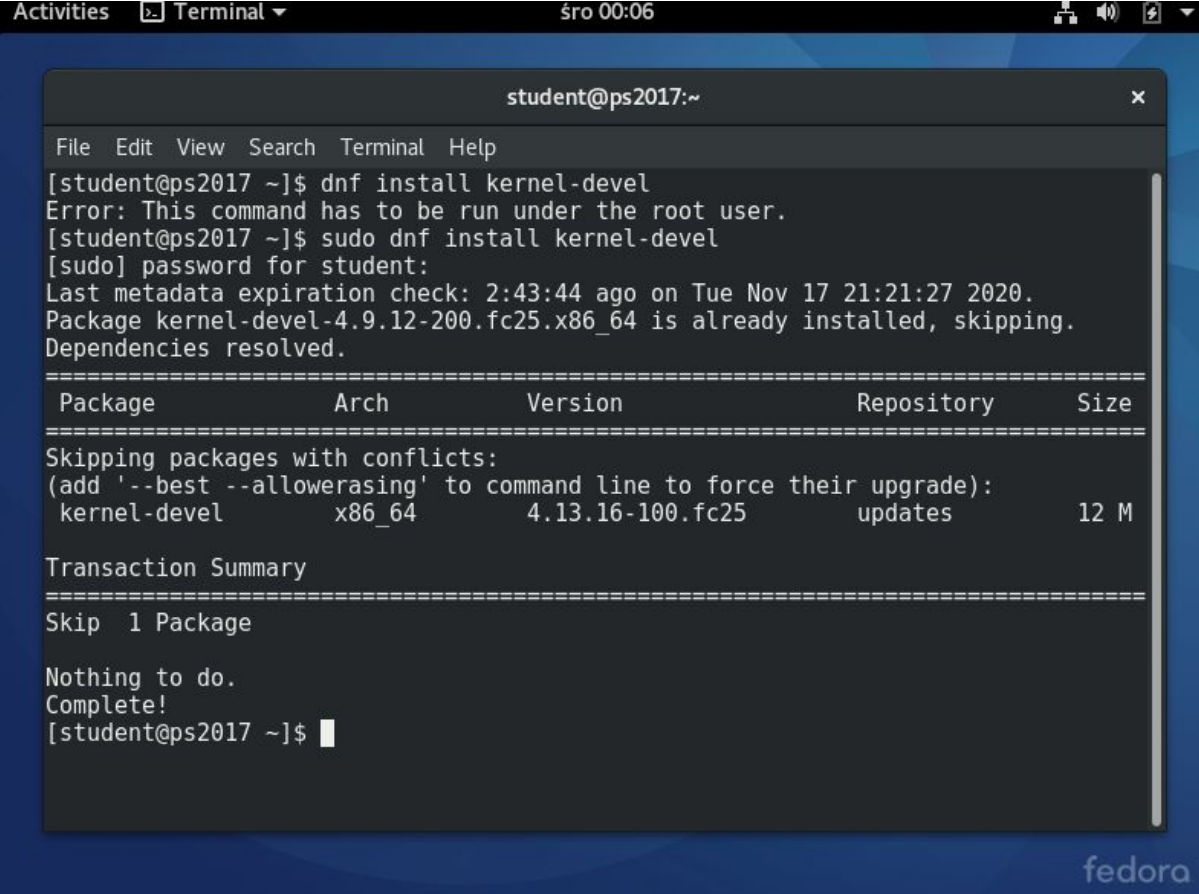
make V=0|1 [targets] 0 => quiet build (default), 1 => verbose build
make V=2 [targets] 2 => give reason for rebuild of target
make O=dir [targets] Locate all output files in "dir", including .config
make C=1 [targets] Check all c source with $CHECK (sparse by default)
make C=2 [targets] Force check of all c source with $CHECK
make RECORDMCOUNT_WARN=1 [targets] Warn about ignored mcount sections
make W=n [targets] Enable extra gcc checks, n=1,2,3 where
               1: warnings which may be relevant and do not occur too often
               2: warnings which occur quite often but may still be relevant
               3: more obscure warnings, can most likely be ignored
               Multiple levels can be combined with W=12 or W=123

Execute "make" or "make all" to build all targets marked with [*]
For further info see the ./README file
[student@ps2017 linux]$ make modules_install
```

```
Activities Terminal wto 19:40
student@ps2017:~/linux
File Edit View Search Terminal Help
INSTALL /lib/firmware/emi62/midi.fw
INSTALL /lib/firmware/kaweth/new_code.bin
INSTALL /lib/firmware/kaweth/trigger_code.bin
INSTALL /lib/firmware/kaweth/new_code_fix.bin
INSTALL /lib/firmware/kaweth/trigger_code_fix.bin
INSTALL /lib/firmware/ti_3410.fw
INSTALL /lib/firmware/ti_5052.fw
INSTALL /lib/firmware/mts_cdma.fw
INSTALL /lib/firmware/mts_gsm.fw
INSTALL /lib/firmware/mts_edge.fw
INSTALL /lib/firmware/edgeport/boot.fw
INSTALL /lib/firmware/edgeport/boot2.fw
INSTALL /lib/firmware/edgeport/down.fw
INSTALL /lib/firmware/edgeport/down2.fw
INSTALL /lib/firmware/edgeport/down3.bin
INSTALL /lib/firmware/whiteheat_loader.fw
INSTALL /lib/firmware/whiteheat.fw
INSTALL /lib/firmware/keyspan_pda/keyspan_pda.fw
INSTALL /lib/firmware/keyspan_pda/xircom_pgs.fw
INSTALL /lib/firmware/cpia2/stv0672_vp4.bin
INSTALL /lib/firmware/yam/1200.bin
INSTALL /lib/firmware/yam/9600.bin
DEPMOD 4.10.0
[student@ps2017 linux]$ sudo make install
```

7. Zrestartuj system i uruchom nowe jądro.

Zadanie 3



```
student@ps2017:~  
File Edit View Search Terminal Help  
[student@ps2017 ~]$ dnf install kernel-devel  
Error: This command has to be run under the root user.  
[student@ps2017 ~]$ sudo dnf install kernel-devel  
[sudo] password for student:  
Last metadata expiration check: 2:43:44 ago on Tue Nov 17 21:21:27 2020.  
Package kernel-devel-4.9.12-200.fc25.x86_64 is already installed, skipping.  
Dependencies resolved.  
=====
```

Package	Arch	Version	Repository	Size
Skipping packages with conflicts: (add '--best --allowrasing' to command line to force their upgrade):				
kernel-devel	x86_64	4.13.16-100.fc25	updates	12 M

```
=====
```

Transaction Summary

=====

Skip 1 Package

Nothing to do.
Complete!
[student@ps2017 ~]\$

Załaduj: insmod.

```
[student@ps2017 trivial_module]$ sudo insmod trivial.ko  
[sudo] password for student:  
[student@ps2017 trivial_module]$
```


Wylistuj moduły: lsmod.

```
su: Authentication failure
[student@ps2017 trivial_module]$ sudo lsmod
Module                Size  Used by
trivial                1163   0
ata_generic            4099   0
vmwgfx                 234236 3
drm_kms_helper         142824 1 vmwgfx
ttm                    88451 1 vmwgfx
drm                    342499 6 vmwgfx,ttm,drm_kms_helper
fjes                   65301  0
crc32c_intel           13519  0
serio_raw              5946   0
pata_acpi              4029   0
e1000                  133100 0
[student@ps2017 trivial_module]$
```

Obejrzyj komunikaty jądra: dmesg

```
[ 33.285620] IPv6: ADDRCONF(NETDEV UP): enp0s3: link is not ready
[ 33.286536] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control:
RX
+ [ 33.286989] IPv6: ADDRCONF(NETDEV CHANGE): enp0s3: link becomes ready
[ 70.138238] systemd-journald[545]: File /var/log/journal/ac4016af79b54d08b298
89c6648e1ab1/user-1000.journal corrupted or uncleanly shut down, renaming and re
placing.
[ 516.285620] trivial: loading out-of-tree module taints kernel.
[ 516.285631] trivial: module verification failed: signature and/or required ke
y missing - tainting kernel
[ 516.285682] Hello world! I'm a trivial module!
[student@ps2017 trivial_module]$
```

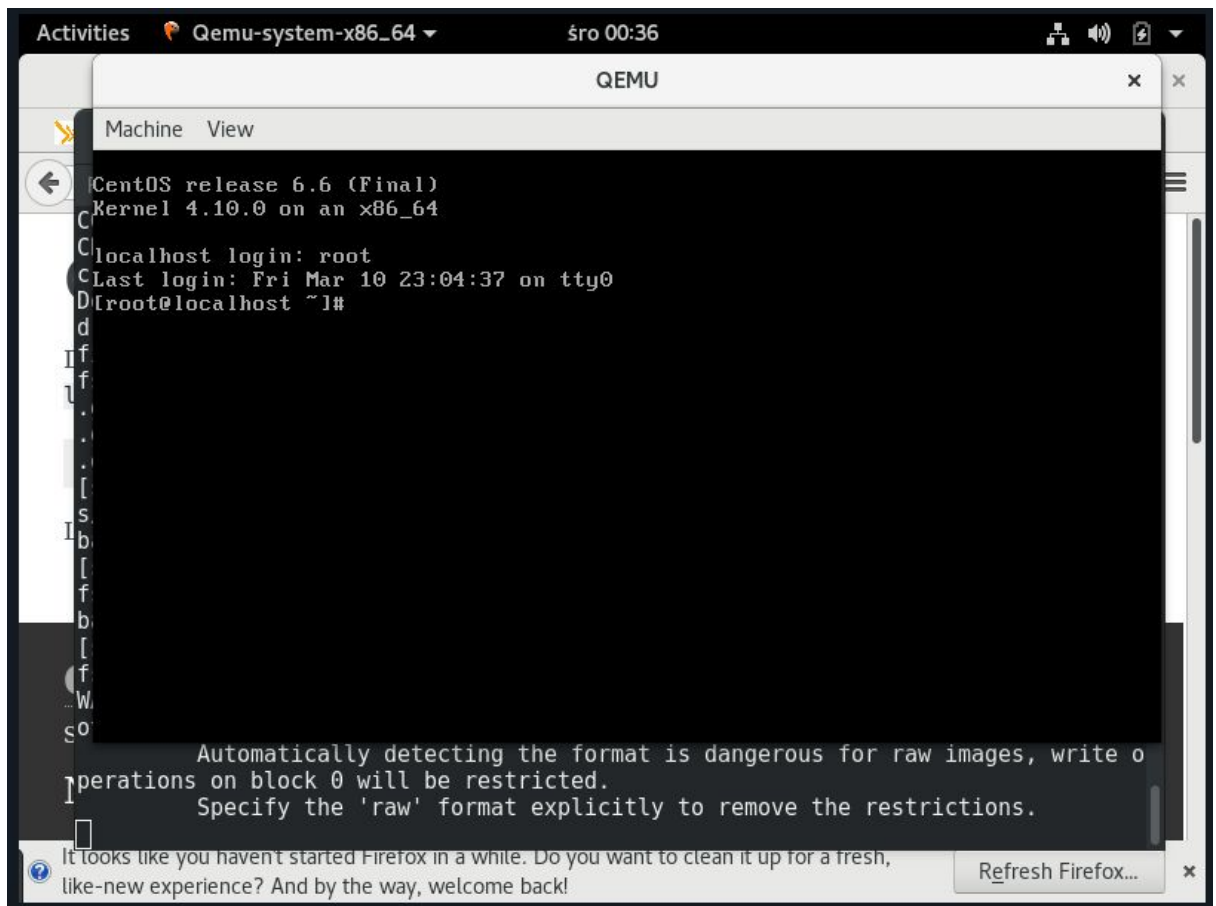
Usuń rmmod.

Jeszcze raz obejrzyj komunikaty jądra.

```
[ 516.285631] trivial: module verification failed: signature and/or required ke
y missing - tainting kernel
[ 516.285682] Hello world! I'm a trivial module!
[ 610.074980] Oh no, why are you doing this to me? Argh!
[student@ps2017 trivial_module]$
```

Zadanie 4

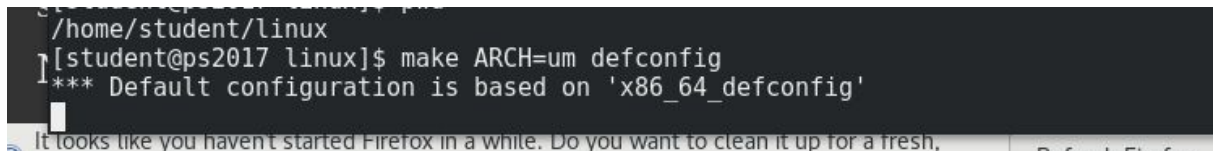
1. Skompiluj jądro, podobnie jak we wcześniejszym zadaniu.
2. Zaloguj się do systemu. Login to *root*, hasło jest puste.



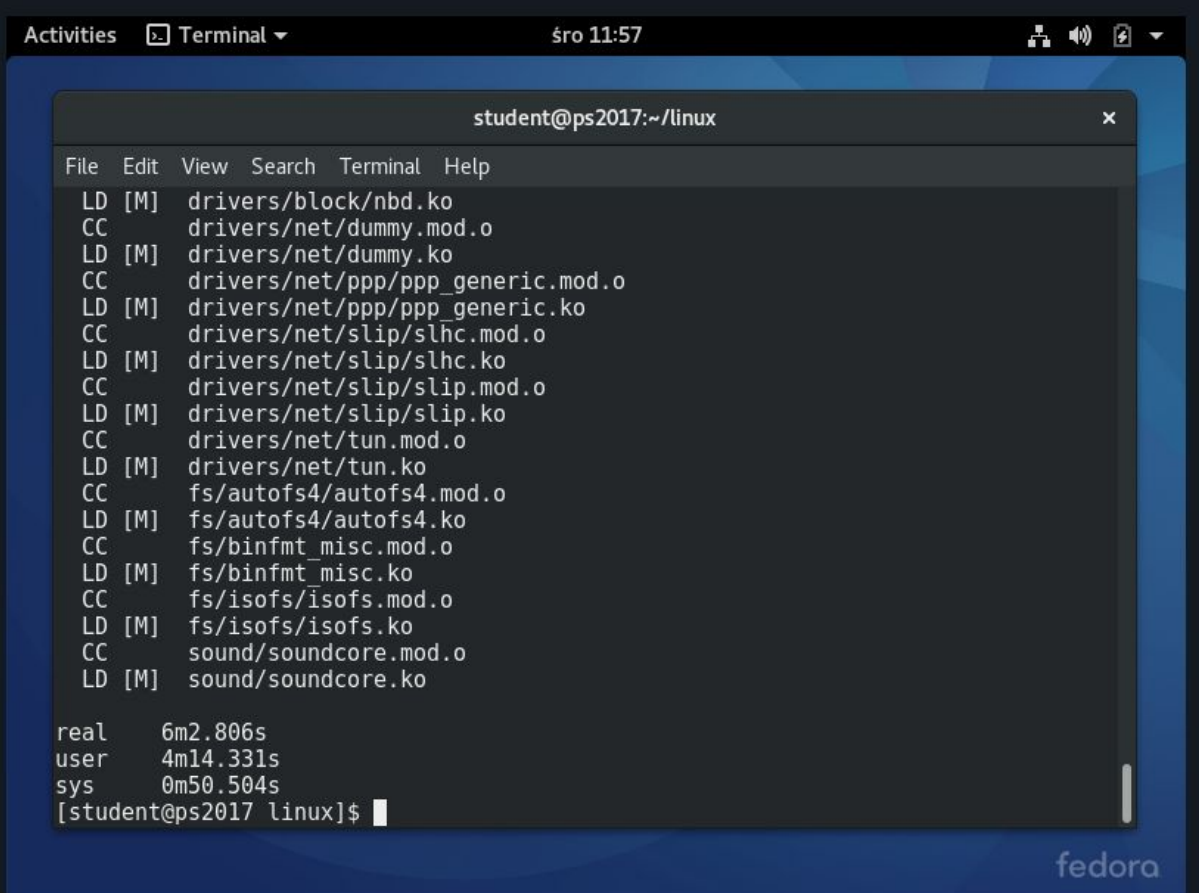
- 3.
4. Zakończ pracę z gościem.

Zadanie 5

1. Wykonaj komendę `make ARCH=um defconfig`. Co ona robi?



3. Skompiluj jądro komendą `make ARCH=um`. Ile czasu zajęła kompilacja?

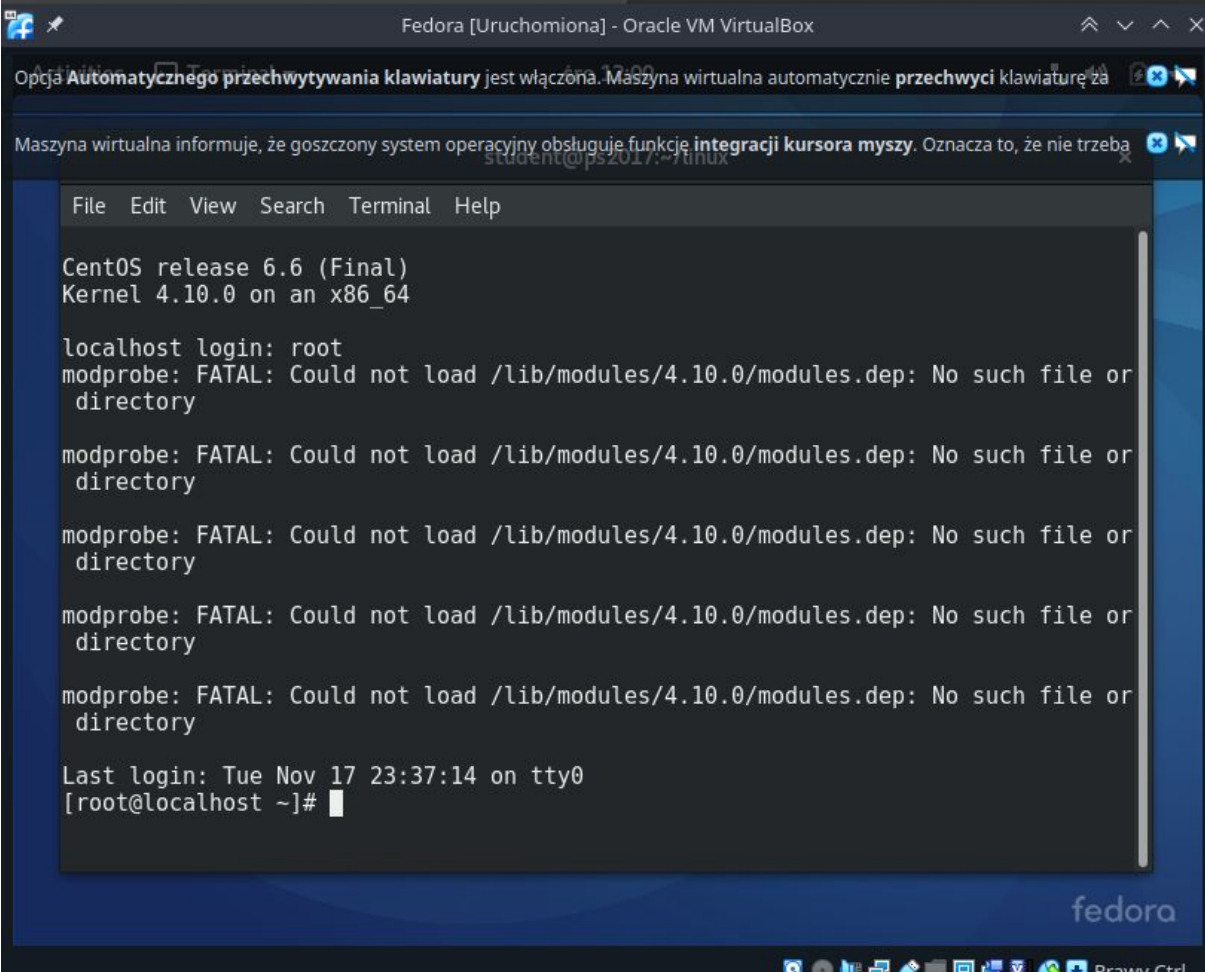


```
student@ps2017:~/linux
File Edit View Search Terminal Help
LD [M] drivers/block/nbd.ko
CC drivers/net/dummy.mod.o
LD [M] drivers/net/dummy.ko
CC drivers/net/ppp/ppp_generic.mod.o
LD [M] drivers/net/ppp/ppp_generic.ko
CC drivers/net/slip/slhc.mod.o
LD [M] drivers/net/slip/slhc.ko
CC drivers/net/slip/slip.mod.o
LD [M] drivers/net/slip/slip.ko
CC drivers/net/tun.mod.o
LD [M] drivers/net/tun.ko
CC fs/autofs4/autofs4.mod.o
LD [M] fs/autofs4/autofs4.ko
CC fs/binfmt_misc.mod.o
LD [M] fs/binfmt_misc.ko
CC fs/isofs/isofs.mod.o
LD [M] fs/isofs/isofs.ko
CC sound/soundcore.mod.o
LD [M] sound/soundcore.ko

real    6m2.806s
user    4m14.331s
sys     0m50.504s
[student@ps2017 linux]$
```

4. Uruchom `./vmlinux ubd0=~/.fs/CentOS6.x-AMD64-root_fs`.

5. Zaloguj się do systemu. Login to *root*, hasło jest puste.



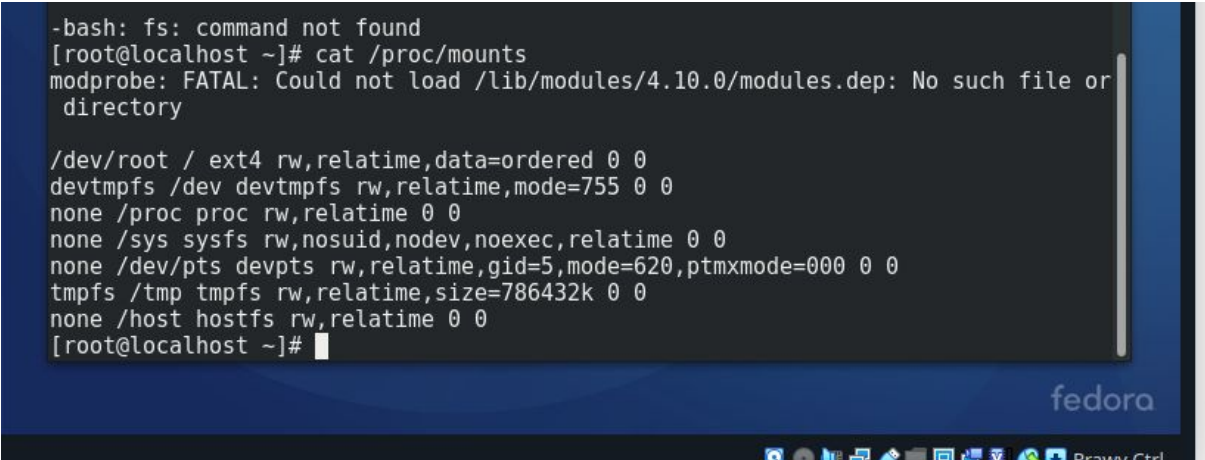
The screenshot shows a terminal window titled "Fedora [Uruchomiona] - Oracle VM VirtualBox". At the top, there are two system messages in Polish: "Opcja Automatycznego przechwytywania klawiatury jest włączona. Maszyna wirtualna automatycznie przechwyci klawiaturę za" and "Maszyna wirtualna informuje, że goszczony system operacyjny obsługuje funkcję integracji kursora myszy. Oznacza to, że nie trzeba". The terminal output shows the system version "CentOS release 6.6 (Final)" and kernel "Kernel 4.10.0 on an x86_64". It then shows a successful login as root: "localhost login: root". Following this, there are several "modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or directory" messages. The last login is noted as "Last login: Tue Nov 17 23:37:14 on tty0". The prompt is "[root@localhost ~]#". The terminal has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The bottom of the window shows the "fedora" logo and a taskbar with various icons and the text "Prawy Ctrl".

```
CentOS release 6.6 (Final)
Kernel 4.10.0 on an x86_64

localhost login: root
modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or
directory

Last login: Tue Nov 17 23:37:14 on tty0
[root@localhost ~]#
```

6. Zamontuj *hostfs* za pomocą komendy: `mount none /host -t hostfs`. (Jeżeli katalog `/host` nie istnieje, to go utwórz.) Co pojawiło się po zamontowaniu tego systemu plików?



The screenshot shows a terminal window with the following output: "-bash: fs: command not found", "[root@localhost ~]# cat /proc/mounts", and then the contents of the /proc/mounts file. The output shows several mounted filesystems, including /dev/root, /devtmpfs, /proc, /sys, /dev/pts, /tmp, and /host. The /host entry is "none /host hostfs rw,relatime 0 0". The prompt is "[root@localhost ~]#". The terminal has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The bottom of the window shows the "fedora" logo and a taskbar with various icons and the text "Prawy Ctrl".

```
-bash: fs: command not found
[root@localhost ~]# cat /proc/mounts
modprobe: FATAL: Could not load /lib/modules/4.10.0/modules.dep: No such file or
directory

/dev/root / ext4 rw,relatime,data=ordered 0 0
devtmpfs /dev devtmpfs rw,relatime,mode=755 0 0
none /proc proc rw,relatime 0 0
none /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
none /dev/pts devpts rw,relatime,gid=5,mode=620,ptmxmode=000 0 0
tmpfs /tmp tmpfs rw,relatime,size=786432k 0 0
none /host hostfs rw,relatime 0 0
[root@localhost ~]#
```


7. Jakie procesy są widoczne w gościu i po stronie hosta?

```
student@ps2017:~/linux
File Edit View Search Terminal Help
root      1  0.1  8.1 19284 2256 ?        Ss   05:59   0:00 /sbin/init
root      2  0.0  0.0      0   0 ?        S    05:59   0:00 [kthreadd]
root      3  0.0  0.0      0   0 ?        S    05:59   0:00 [kworker/0:0]
root      4  0.0  0.0      0   0 ?        S<   05:59   0:00 [kworker/0:0H]
root      5  0.0  0.0      0   0 ?        S    05:59   0:00 [kworker/u2:0]
root      6  0.0  0.0      0   0 ?        S    05:59   0:00 [ksoftirqd/0]
root      7  0.0  0.0      0   0 ?        S<   05:59   0:00 [lru-add-drain]
root      8  0.0  0.0      0   0 ?        S    05:59   0:00 [kdevtmpfs]
root      9  0.0  0.0      0   0 ?        S<   05:59   0:00 [netns]
root     10  0.0  0.0      0   0 ?        S    05:59   0:00 [oom reaper]
root     11  0.0  0.0      0   0 ?        S    05:59   0:00 [kworker/u2:1]
root     45  0.0  0.0      0   0 ?        S<   05:59   0:00 [writeback]
root     46  0.0  0.0      0   0 ?        S<   05:59   0:00 [crypto]
root     47  0.0  0.0      0   0 ?        S    05:59   0:00 [kworker/0:1]
root     49  0.0  0.0      0   0 ?        S<   05:59   0:00 [bioset]
root     51  0.0  0.0      0   0 ?        S<   05:59   0:00 [kblockd]
root     73  0.0  0.0      0   0 ?        S    05:59   0:00 [kswapd0]
root     74  0.0  0.0      0   0 ?        S<   05:59   0:00 [bioset]
root    227  0.0  0.0      0   0 ?        S<   05:59   0:00 [bioset]
root    236  0.0  0.0      0   0 ?        S    05:59   0:00 [jbd2/ubda-8]
root    237  0.0  0.0      0   0 ?        S<   05:59   0:00 [ext4-rsv-conv]
root    305  0.0  5.3 10700 1472 ?        S<S   05:59   0:00 /sbin/udevd -d
root    400  0.0  5.1 10696 1416 ?        S<   05:59   0:00 /sbin/udevd -d
root    406  0.0  0.0      0   0 ?        S<   05:59   0:00 [kworker/0:1H]
root    576  0.0  9.0 66268 2500 ?        Ss   05:59   0:00 /usr/sbin/sshd
root    589  0.7  9.0 52352 2492 ?        Ss   05:59   0:01 login -- root
root    595  0.0  9.5 11488 2628 tty0      Ss   06:00   0:00 -bash
```

```
student@ps2017:~/linux
File Edit View Search Terminal Help
[student@ps2017 linux]$ ps aux | grep vmlinux
student 11846 1.4 0.5 45552 33740 pts/0    S+   11:59   0:06 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 11851 0.0 0.5 45552 33740 pts/0    S+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 11852 0.0 0.5 45552 33740 pts/0    S+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 11853 0.0 0.5 45552 33740 pts/0    S+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 11854 0.0 0.0 14908 2348 pts/0    t+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 11984 0.0 0.0 14124 732 pts/0    t+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 12128 0.0 0.0 14068 792 pts/0    t+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 12428 0.0 0.0 13844 756 pts/0    t+   11:59   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 12453 0.0 0.0 15144 1356 pts/0    t+   12:00   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 12460 0.0 0.0 15280 1860 pts/0    t+   12:00   0:00 ./vmlinux ubd0=/home/s
student/fs/CentOS6.x-AMD64-root fs
student 12565 0.0 0.0 119372 900 pts/1    S+   12:06   0:00 grep --color=auto vml
nux
[student@ps2017 linux]$
root      606  0.0  0.0      0   0 ?        S    06:01   0:00 [kworker/u2:2]
root      620  7.0  6.7 13372 1860 tty0      R+   06:02   0:00 ps aux
[root@localhost ~]#
```

8. Zakończ pracę z gościem.